文章编号: 1001-0920 (2013) 08-1121-09

高斯过程回归方法综述

何志昆, 刘光斌, 赵曦晶, 王明昊

(第二炮兵工程大学 控制工程系，西安 710025)

摘 要: 高斯过程回归是基于贝叶斯理论和统计学习理论发展起来的一种全新机器学习方法, 适于处理高维数、小样本和非线性等复杂回归问题. 在阐述该方法原理的基础上, 分析了其存在的计算量大、噪声必须服从高斯分布等问题, 给出了改进方法. 与神经网络和支持向量机相比, 该方法具有容易实现、超参数自适应获取以及输出具有概率意义等优点, 方便与预测控制、自适应控制、贝叶斯滤波等相结合. 最后总结了其应用情况并展望了未来发展方向.

关键词: 高斯过程回归；机器学习；函数空间；协方差矩阵；近似法；不确定度

中图分类号: TP181 **文献标志码:** A

Overview of Gaussian process regression

HE Zhi-kun, *LIU Guang-bin*, *ZHAO Xi-jing*, *WANG Ming-hao*

(Department of Control Engineering，The Second Artillery Engineering University，Xi'an 710025，China. Correspondent：HE Zhi-kun，E-mail：hezhikun0@sina.com)

Abstract: Gaussian process regression(GPR) is a new machine learning method by the context of Bayesian theory and statistical learning theory. It provides a flexible framework for probabilistic regression and is widely used to solve the high-dimensional, small-sample or nonlinear regression problems. Its principle is introduced in the function-space view and several limitations such as computational difficulties for large data sets and restrictive modelling assumptions for complex data sets are discussed. Several improved approaches for these limitations are summarized. GPR is simple to implement, flexible to nonparameter infer and self-adaptive to determinate hyperparameters in comparison with neural network and support vector machines. The attractive feature that GPR models provide Gaussian uncertainty estimates for their predictions allows them to be seamlessly incorporated into predictive control, adaptive control and Bayesian filtering techniques. Finally, its applications are given and future research trends are prospected.

Key words: Gaussian process regression; machine learning; function space; covariance matrix; approximations; uncertainty

0 引 言

机器学习是当前计算机科学和信息科学中一个重要的前沿领域, 与模式识别和统计推断密切相关, 正逐渐为各领域学者所重视. 它是一门多学科交叉研究, 研究内容和应用领域极其广泛, 几乎囊括了所有人类认知领域. 机器学习问题大体可以分为三大类: 监督学习、无监督学习和强迫学习. 根据经验数据 (训练集) 来学习输入-输出之间的映射关系, 使得给定新的输入便可得到相应的输出值(即预测值), 即为监督学习问题. 根据输出值的类型, 可以分为回归问题 (输出为连续的) 和分类问题 (输出为离散的). 其中, 回归问题可以数学描述如下:

假设有训练集 $D = \{(\boldsymbol{x}_i, y_i)|i = 1, 2, \cdots, n\} = (X, \boldsymbol{y})$. 其中: $\boldsymbol{x}_i \in R^d$ 为 d 维输入矢量, $X = [\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n]$ 为 $d \times n$ 维输入矩阵, $y_i \in R$ 为相应的输出标量, $\boldsymbol{y}$ 为输出矢量. 回归的任务是根据训练集学习输入 X 与输出 $\boldsymbol{y}$ 之间的映射关系 ($f(\cdot) : R^d \mapsto R$), 预测出与新测试点 $\boldsymbol{x}_*$ 对应的最可能输出值 $f(\boldsymbol{x}_*)$.

在监督学习中, 通常采用两类方法来确定映射函数. 第 1 类是参数化回归, 即假设训练数据是通过一个由参数 $\boldsymbol{w}$ 定义的函数 $f(\boldsymbol{x}; \boldsymbol{w})$ 产生得到的. 此时, 函数映射 $f(\boldsymbol{x}; \cdot)$ 和特定参数集 $\boldsymbol{w}$ 共同定义了参数化模型, 而参数化回归即是寻找一组使数据得到"最好"诠释的参数. 该方法引入了一个新的问题: 如何判

收稿日期: 2012-10-09；**修回日期:** 2012-12-17.

基金项目: 国家863计划项目(2010AA7010213).

作者简介: 何志昆(1984−), 男, 博士生, 从事机器学习、非线性滤波及组合导航的研究；刘光斌(1963−), 男, 教授, 博士生导师, 从事系统辨识与仿真、卫星信号仿真等研究.

断一个模型是最好的, 或者一个模型比另一个模型更好? 一种方法是寻找一组能使某一损失函数 $L(\boldsymbol{w})$ 最小化的参数. 通常采用的损失函数为二次损失函数, 典型的例子有最小二乘多项式回归、最小二乘BP神经网络等. 这种方法存在明显的缺陷: 仅致力于在训练集上降低模型误差. 若为了降低模型误差而一味增加模型复杂度, 则易导致过拟合, 尽管在训练集上回归精度较高, 但其泛化能力或预测性能不佳. 为了避免过拟合, 可以使用一个相对简单的模型, 它忽略了复杂特征和噪声, 相对比较平滑. 但是模型过于简单也会造成预测性能差. 另一种方法是极大似然法, 它不需要损失函数. 首先由假定的噪声分布得到训练集的联合概率密度(即似然函数), 再通过寻找使似然函数最大化的参数 $\boldsymbol{w}$ 来获得回归模型. 如果噪声分布满足高斯分布, 则通过比较似然函数和二次损失函数不难发现, 该似然函数的负对数与二次损失函数成一定比例关系, 因而表明了这两种方法在本质上是一样的.

为了避免过拟合, 可以采用第2类方法, 即贝叶斯回归. 该方法定义了一个函数分布, 赋予每一种可能的函数一个先验概率, 可能性越大的函数, 其先验概率越大. 但是可能的函数往往为一个不可数集, 即有无限个可能的函数. 随之引入一个新的问题: 如何在有限的时间内对这些无限的函数进行选择? 一种有效的解决方法即是高斯过程回归(GPR).

GPR是近年发展起来的一种机器学习回归方法, 它有着严格的统计学习理论基础, 对处理高维数、小样本、非线性等复杂的问题具有很好的适应性, 且泛化能力强. 与神经网络、支持向量机相比, GPR具有容易实现、超参数自适应获取、非参数推断灵活以及输出具有概率意义等优点, 在国外发展很快, 并取得了许多研究成果, 现已成为国际机器学习领域的研究热点[1-3]; 近几年也逐步得到国内学者的重视, 在许多领域得到了成功应用[4-6]. 本文将首先阐述GPR的基本原理, 对GPR存在的主要问题进行探讨, 总结了相应的改进方法. 最后对GPR的应用进行了总结并指出其未来发展趋势.

1 高斯过程回归原理

1.1 预 测

从函数空间角度出发, 定义一个高斯过程(GP)来描述函数分布, 直接在函数空间进行贝叶斯推理[1,7]. GP是任意有限个随机变量均具有联合高斯分布的集合, 其性质完全由均值函数和协方差函数确定, 即

$$\begin{cases} m(\boldsymbol{x}) = \boldsymbol{E}[f(\boldsymbol{x})], \\ k(\boldsymbol{x},\boldsymbol{x'}) = \boldsymbol{E}[(f(\boldsymbol{x})-m(\boldsymbol{x}))(f(\boldsymbol{x'})-m(\boldsymbol{x'}))], \end{cases}$$

其中 $\boldsymbol{x},\boldsymbol{x'} \in R^d$ 为任意随机变量. 因此GP可定义为 $f(\boldsymbol{x}) \sim \mathrm{GP}(m(\boldsymbol{x}), k(\boldsymbol{x},\boldsymbol{x'}))$. 为了符号上的简洁, 通常对数据作预处理, 使其均值函数等于0.

对于回归问题, 考虑如下模型:

$$y = f(\boldsymbol{x}) + \varepsilon. \tag{1}$$

其中: $\boldsymbol{x}$ 为输入向量, f 为函数值, y 为受加性噪声污染的观测值, 进一步假设噪声 $\varepsilon \sim N(0,\sigma_n^2)$. 可以得到观测值 $\boldsymbol{y}$ 的先验分布为

$$\boldsymbol{y} \sim N(0, K(X,X) + \sigma_n^2 I_n),$$

以及观测值 $\boldsymbol{y}$ 和预测值 f_* 的联合先验分布为

$$\begin{bmatrix} \boldsymbol{y} \\ f_* \end{bmatrix} \sim N\left(0, \begin{bmatrix} K(X,X)+\sigma_n^2 I_n & K(X,\boldsymbol{x}_*) \\ K(\boldsymbol{x}_*,X) & k(\boldsymbol{x}_*,\boldsymbol{x}_*) \end{bmatrix}\right).$$

其中: $K(X,X) = K_n = (k_{ij})$ 为 $n\times n$ 阶对称正定的协方差矩阵, 矩阵元素 $k_{ij} = k(\boldsymbol{x}_i,\boldsymbol{x}_j)$ 用来度量 $\boldsymbol{x}_i$ 和 $\boldsymbol{x}_j$ 之间的相关性; $K(X,\boldsymbol{x}_*) = K(\boldsymbol{x}_*,X)^{\mathrm{T}}$ 为测试点 $\boldsymbol{x}_*$ 与训练集的输入 X 之间的 $n\times 1$ 阶协方差矩阵; $k(\boldsymbol{x}_*,\boldsymbol{x}_*)$ 为测试点 $\boldsymbol{x}_*$ 自身的协方差; I_n 为 n 维单位矩阵.

由此可以计算出预测值 f_* 的后验分布为

$$f_*|X,\boldsymbol{y},\boldsymbol{x}_* \sim N(\bar{f}_*, \mathrm{cov}(f_*)).$$

其中

$$\bar{f}_* = K(\boldsymbol{x}_*,X)[K(X,X)+\sigma_n^2 I_n]^{-1}\boldsymbol{y}, \tag{2}$$

$$\mathrm{cov}(f_*) = k(\boldsymbol{x}_*,\boldsymbol{x}_*) - K(\boldsymbol{x}_*,X)\times [K(X,X)+\sigma_n^2 I_n]^{-1}K(X,\boldsymbol{x}_*). \tag{3}$$

则 $\hat{\mu}_* = \bar{f}_*$, $\hat{\sigma}_{f_*}^2 = \mathrm{cov}(f_*)$ 即为测试点 $\boldsymbol{x}_*$ 对应预测值 f_* 的均值和方差.

1.2 训 练

GPR可以选择不同的协方差函数, 常用的协方差函数有平方指数协方差, 即

$$k(\boldsymbol{x},\boldsymbol{x'}) = \sigma_f^2 \exp\Big(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{x'})^{\mathrm{T}}M^{-1}(\boldsymbol{x}-\boldsymbol{x'})\Big).$$

其中: $M = \mathrm{diag}(l^2)$, l 为方差尺度, σ_f^2 为信号方差. 参数集合 $\boldsymbol{\theta} = \{M,\sigma_f^2,\sigma_n^2\}$ 即为超参数, 一般通过极大似然法求得. 首先建立训练样本条件概率的负对数似然函数 $L(\boldsymbol{\theta}) = -\log p(\boldsymbol{y}|X,\boldsymbol{\theta})$, 并令其对超参数 $\boldsymbol{\theta}$ 求偏导; 然后采用共轭梯度法、牛顿法等优化方法对偏导数进行最小化以得到超参数的最优解. 这里, 负对数似然函数 $L(\boldsymbol{\theta})$ 及其关于超参数 $\boldsymbol{\theta}$ 的偏导数形式如下所示:

$$L(\boldsymbol{\theta}) = \frac{1}{2}\boldsymbol{y}^{\mathrm{T}}C^{-1}\boldsymbol{y} + \frac{1}{2}\log|C| + \frac{n}{2}\log 2\pi,$$

$$\frac{\partial L(\boldsymbol{\theta})}{\partial \theta_i} = \frac{1}{2}\mathrm{tr}\left((\boldsymbol{\alpha}\boldsymbol{\alpha}^{\mathrm{T}} - C^{-1})\frac{\partial C}{\partial \theta_i}\right).$$

其中

$$C = K_n + \sigma_n^2 I_n,\ \boldsymbol{\alpha} = (K+\sigma_n^2 I_n)^{-1}\boldsymbol{y} = C^{-1}\boldsymbol{y}.$$

获得最优超参数后, 利用式 (2) 和 (3) 便可得到测试点 $\boldsymbol{x}_*$ 对应的预测值 f_* 及其方差 $\hat{\sigma}^2_{f_*}$.

2　GPR 存在的主要问题及改进方法

尽管 GPR 方法具有容易实现、超参数自适应获取以及预测输出具有概率意义等优点, 但是它目前仍存在一些问题, 主要有两个方面: 一是计算量大; 二是局限于高斯噪声分布假设.

2.1　降低计算量的改进方法

GPR 的非参数性质直接导致其计算量大的问题. 如前所述, 训练中超参数一般是通过最优化边缘似然获取的. 每一次梯度计算都需要对协方差矩阵 $K_n + \sigma_n^2 I_n$ 求逆, 因此计算量为 $O(n^3\times$ 梯度计算的次数). 预测时, 每个测试点的预测计算量为 $O(n^2)$. 当处理大数据集时, 计算量将成为限制高斯过程回归方法应用的一大瓶颈.

过去 20 年里, 为了解决上述问题, 人们做了大量的工作, 提出了许多有效的近似方法, 大体上可以分为以下 3 类.

2.1.1　数据子集 (SD) 近似法

在众多降低计算复杂度的方法中, 最简单的便是 SD 近似法 —— 仅选择原 n 维训练集中的一个维数为 m 的小子集作为新训练集, 用于 GPR 预测. 尽管该方法看似简单, 但是相对于其他更复杂的近似方法而言, 它没有额外的计算量和内存开销, 在许多场合下可能是最好的方法: 例如对于高度冗余数据集而言, 额外的数据点能提供关于函数的信息非常少, 此时没有必要牺牲计算量来采用其他复杂的近似方法以获得性能上微不足道的改善. 应用 SD 近似法的关键是如何选取一个合适的数据子集. 下文的许多算法也都面临同样的问题. 目前通常采用的方法有两种: 1) 随机选取; 2) 贪心算法 (greedy approach), 也称前向选取策略 (forward selection strategy).

2.1.2　降秩 (reduced-rank) 近似法

降低计算量的另一种思路是对协方差矩阵 K_n 进行降秩近似, 即 $K_n = VV^{\mathrm{T}}$, 其中 V 为 $n\times m$ 维 $(m<n)$ 矩阵. 此时, 由矩阵求逆引理可得

$$(K_n + \sigma_n^2 I_n)^{-1} = \sigma_n^{-2} I_n - \sigma_n^{-2} V(\sigma_n^2 I_p + V^{\mathrm{T}}V)^{-1}V^{\mathrm{T}}.$$

从上式可以看出, $n\times n$ 维矩阵的求逆已经转变成 $m\times m$ 维矩阵的求逆, 训练计算量已由 $O(n^3)$ 降至 $O(n^2m)$, 预测计算量由 $O(n^2)$ 降至 $O(m^2)$. 但是如何实现 $K_n = VV^{\mathrm{T}}$ 是该方法的关键. 采用特征值分解, 然后保留 m 个主导特征值的方法可以实现该步骤, 但是由于一般情况下对 K_n 进行特征值分解的计算量同样高达 $O(n^3)$, 可见该方法不适用. 于是可以采用高效 (计算量小) 的近似特征值分解方法, 其中应用较广泛的是 Nyström 方法[8].

1) Nyström 近似法.

类似于 SD 近似法, 从原训练集中选取一个维数为 m 的子集, 称为包含集或活动集, 则 K_n 可模块分解为

$$K_n = \begin{bmatrix} K_{mm} & K_{m(n-m)} \\ K_{(n-m)m} & K_{(n-m)(n-m)} \end{bmatrix},$$

上式顶部 $m\times n$ 模块记为 K_{mn} (其转置为 K_{nm}). 采用 Nyström 方法构建 K_n, 得到一个近似协方差矩阵

$$\tilde{K}_n = K_{nm}K_{mm}^{-1}K_{mn},$$

此时, $\tilde{K}_n$ 的计算量为 $O(m^2n)$. 同时可得

$$K_{mm} = \tilde{K}_{mm},$$
$$K_{m(n-m)} = \tilde{K}_{m(n-m)},$$
$$\tilde{K}_{(n-m)(n-m)} = K_{(n-m)m}K_{mm}^{-1}K_{m(n-m)},$$
$$K_{(n-m)m} = \tilde{K}_{(n-m)m}.$$

记 $k_m(\boldsymbol{x}) = [k(\boldsymbol{x},\boldsymbol{x}_1),\cdots,k(\boldsymbol{x},\boldsymbol{x}_m)]^{\mathrm{T}}$, $\tilde{k}(\boldsymbol{x},\boldsymbol{x}') = k_m(\boldsymbol{x})^{\mathrm{T}}K_{mm}^{-1}k_m(\boldsymbol{x}')$. Williams 等[8]直接在式 (2) 和 (3) 中用 $\tilde{K}_n$ 替换 K_n, 该方法称为 GPR 的 Nyström 近似法. 它的训练计算量降至 $O(m^2n)$, 单测试样本的均值和协方差预测计算量分别降至 $O(n)$ 和 $O(mn)$.

2) 回归量子集 (SR) 法.

式 (2) 还可写成如下形式:

$$\hat{\mu}_* = \sum_{i=1}^{n}\alpha_i k(\boldsymbol{x}_i,\boldsymbol{x}_*), \tag{4}$$

其中 $\boldsymbol{\alpha}_i$ 为 $\boldsymbol{\alpha} = [K(X,X)+\sigma_n^2 I_n]^{-1}\boldsymbol{y}$ 的第 i 个元素. 由式 (4) 可得, 一个简单的近似即是仅考虑回归量的一个子集, 即

$$f_{\mathrm{SR}}(\boldsymbol{x}_*) = \sum_{i=1}^{m}\alpha_i k(\boldsymbol{x}_i,\boldsymbol{x}_*),$$

其中 $\alpha_m \sim N(0,K_{mm}^{-1})$. 该方法最早是由 Wahba[9]和 Poggio 等[10]提出, 并由 Wahba 将其命名为 SR 近似法. SR 近似法与 Nyström 近似法的不同之处在于它用 $\tilde{k}(\boldsymbol{x},\boldsymbol{x}')$ 替代式 (2) 和 (3) 中的 $k(\boldsymbol{x},\boldsymbol{x}')$, 得到

$$\bar{f}_{\mathrm{SR}}(\boldsymbol{x}_*) = k_m(\boldsymbol{x}_*)^{\mathrm{T}}(K_{mn}K_{nm}+\sigma_n^2K_{mm})^{-1}K_{mn}\boldsymbol{y},$$
$$\mathrm{cov}(f_{\mathrm{SR}}(\boldsymbol{x}_*)) = \sigma_n^2 k_m(\boldsymbol{x}_*)^{\mathrm{T}}(K_{mn}K_{nm}+\sigma_n^2K_{mm})^{-1}k_m(\boldsymbol{x}_*).$$

SR 近似法的训练计算量为 $O(m^2n)$, 单测试样本的均值和协方差预测计算量分别为 $O(m)$ 和 $O(m^2)$. 实践证明, 当 m 较大时, SR 近似法和 Nyström 近似法的性能相近; 当 m 较小时, Nyström 近似法的性能将变得非常差[11]. 这是由于 Nyström 近似法不是利用 $\tilde{k}(\boldsymbol{x},\boldsymbol{x}')$ 来代替 $k(\boldsymbol{x},\boldsymbol{x}')$, 可能会导致出现近似预测方差为负的情况, 可见 Nyström 近似法仅适用于当 K_n

的第 $m+1$ 个特征值远远小于 σ_n^2 的情况. 若对于固定的 $\boldsymbol{x}'$, 当 $|\boldsymbol{x}| \to \infty$ 时, $k(\boldsymbol{x}, \boldsymbol{x}') \to 0$, 则当 $\boldsymbol{x}$ 远离包含集时, $\tilde{k}(\boldsymbol{x}, \boldsymbol{x}) \approx 0$, 这导致了预测性能非常差, 特别是低估了预测方差.

3) 映射过程 (PP) 近似法.

SR 近似法得到的结果是一个退化的高斯过程回归模型 (即有限维模型), 而 SD 近似法的不足在于它仅使用了 m 个数据点. 于是有了另一种近似法 —— PP 近似法[12-13], 它利用所有 n 个数据点的信息, 得到了一个非退化的 GPR 模型. 之所以称其为 PP 近似法, 是因为它在计算似然函数时将 $m(< n)$ 个潜在数据点映射到 n 维空间, 从而包含了所有 n 个数据. 令 $\boldsymbol{f}_m$ 表示所选取 m 个数据点的函数值向量, $\boldsymbol{f}_{n-m}$ 为剩余数据点的函数值向量, 则条件概率分布 $p(\boldsymbol{f}_{n-m}|\boldsymbol{f}_m)$ 的均值为 $E[\boldsymbol{f}_{n-m}|\boldsymbol{f}_m] = K_{(n-m)m}K_{mm}^{-1}\boldsymbol{f}_m$. 用 $N(\boldsymbol{y}_{n-m}|E[\boldsymbol{f}_{n-m}|\boldsymbol{f}_m], \sigma_n^2 I)$ 来代替剩余数据点集的真实似然函数, 得到

$$\boldsymbol{y}|\boldsymbol{f}_m \sim N(K_{nm}K_{mm}^{-1}\boldsymbol{f}_m, \sigma_n^2 I) = N(E[\boldsymbol{f}|\boldsymbol{f}_m], \sigma_n^2 I).$$

由上式可以看到, 与 SD 近似法、SR 近似法不同, PP 近似法是将所有 n 个数据点的信息压缩合并到所选取的 m 个数据点中. 从而得到后验分布为

$$\boldsymbol{f}_m|\boldsymbol{y} \sim N(\mu_{\boldsymbol{f}_m|\boldsymbol{y}}, A_{\boldsymbol{f}_m|\boldsymbol{y}}).$$

其中

$$\mu_{\boldsymbol{f}_m|\boldsymbol{y}} = K_{mm}(\sigma_n^2 K_{mm} + K_{mn}K_{nm})^{-1}K_{mn}\boldsymbol{y},$$

$$A_{\boldsymbol{f}_m|\boldsymbol{y}}^{-1} = \sigma_n^{-2}K_{mm}^{-1}(\sigma_n^2 K_{mm} + K_{mn}K_{nm})K_{mm}^{-1}.$$

最终得到

$$\bar{f}_{\rm PP}(\boldsymbol{x}_*) = k_m(\boldsymbol{x}_*)^{\rm T}K_{mm}^{-1}\mu =$$

$$k_m(\boldsymbol{x}_*)^{\rm T}(K_{mn}K_{nm} + \sigma_n^2 K_{mm})^{-1}K_{mn}\boldsymbol{y} =$$

$$\bar{f}_{\rm SR}(\boldsymbol{x}_*),$$

$$\mathrm{cov}(f_{\rm PP}(\boldsymbol{x}_*)) =$$

$$k(\boldsymbol{x}_*, \boldsymbol{x}_*) - k_m(\boldsymbol{x}_*)^{\rm T}K_{mm}^{-1}k_m(\boldsymbol{x}_*) +$$

$$\sigma_n^2 k_m(\boldsymbol{x}_*)^{\rm T}(K_{mn}K_{nm} + \sigma_n^2 K_{mm})^{-1}k_m(\boldsymbol{x}_*) =$$

$$\mathrm{cov}(f_*|\boldsymbol{f}_m) + \mathrm{cov}(f_{\rm SR}(\boldsymbol{x}_*)).$$

可以看出, PP 近似法的预测均值与 SR 近似法相同, 预测方差比 SR 近似法多一项条件预测方差 $\mathrm{cov}(f_*|\boldsymbol{f}_m) = k(\boldsymbol{x}_*, \boldsymbol{x}_*) - k_m(\boldsymbol{x}_*)^{\rm T}K_{mm}^{-1}k_m(\boldsymbol{x}_*)$, 即 $\mathrm{cov}(f_{\rm PP}(\boldsymbol{x}_*)) > \mathrm{cov}(f_{\rm SR}(\boldsymbol{x}_*))$, 且当测试点 $\boldsymbol{x}_*$ 远离所选数据集时, $\mathrm{cov}(f_{\rm PP}(\boldsymbol{x}_*)) \to k(\boldsymbol{x}_*, \boldsymbol{x}_*)$, 避免了 SR 近似法低估预测方差的问题. PP 近似法的训练计算量为 $O(m^2 n)$, 单测试样本的均值和协方差预测计算量分别为 $O(m)$ 和 $O(m^2)$.

此外, 还有许多其他的近似方法, 如 Tresp 等[14] 基于分块数据集提出了 BCM (Bayesian committee machine) 方法用于提高 GPR 效率, 等等.

2.1.3 稀疏伪输入 (SPGP) 法

在前述近似方法中普遍存在一个问题: 由于需要重复选择活动点集和最优化超参数, 而新点集干扰了超参数的最优化, 可能导致收敛困难, 参数学习结果可靠性降低. Snelson 等[15]提出了 SPGP 法. 该方法的主要思想是: 将伪输入集初始化为训练点集的一个子集, 它们是连续变量, 其值通过最优化得到. 这使得 GPR 超参数和伪输入集位置的最优化可以同时进行.

由式 (2) 和 (3) 知, 可将该预测分布的均值和方差分别看作新测试样本 $\boldsymbol{x}_*$ 的函数. 假定超参数已知且固定, 则这两个函数中的参数由训练集 D 中 n 个输入输出对的位置决定. SPGP 法利用一组伪数据集 $\bar{D} = (\bar{X}, \bar{\boldsymbol{f}})$ 来代替真实数据集 D, 并将该伪数据集得到的 GPR 预测分布作为一个参数化的模型似然函数, 其中横杆表示伪数据集不是真实的观测数据, 伪输入 $\bar{X} = \{\bar{\boldsymbol{x}}_i\}_{i=1}^m$, 伪输出 $\bar{\boldsymbol{f}} = \{\bar{f}_i\}_{i=1}^m$ 等价于不含噪声的潜在函数变量值. 而实际观测输出值仍假定受噪声污染 (见式 (1)), 可得

$$y|\boldsymbol{x}_*, \bar{X}, \bar{\boldsymbol{f}} \sim N(\bar{k}_m(\boldsymbol{x}_*)^{\rm T}\bar{K}_{mm}^{-1}\bar{\boldsymbol{f}}, \bar{k}(\boldsymbol{x}_*, \boldsymbol{x}_*) - \bar{k}_m(\boldsymbol{x}_*)^{\rm T}\bar{K}_{mm}^{-1}\bar{k}_m(\boldsymbol{x}_*) + \sigma_n^2).$$

其中: $[\bar{k}_m(\boldsymbol{x}_*)]_i = K(\bar{\boldsymbol{x}}_i, \boldsymbol{x}_*)$, $[\bar{K}_{mm}]_{ij} = K(\bar{\boldsymbol{x}}_i, \bar{\boldsymbol{x}}_j)$. 此时, 可将其看作一个均值函数为特定参数化形式且输入相关的标准回归模型. 应用标准 GPR 原理, 可得

$$\bar{f}_{\rm SP}(\boldsymbol{x}_*) = \bar{k}_m(\boldsymbol{x}_*)^{\rm T}\bar{Q}_{mm}^{-1}\bar{K}_{mn}(\Lambda + \sigma_n^2 I)^{-1}\boldsymbol{y},$$

$$\mathrm{cov}(f_{\rm SP}(\boldsymbol{x}_*)) = k(\bar{\boldsymbol{x}}_*, \bar{\boldsymbol{x}}_*) - \bar{k}_m(\boldsymbol{x}_*)^{\rm T} \times (\bar{K}_{mm}^{-1} - \bar{Q}_{mm}^{-1})\bar{k}_m(\boldsymbol{x}_*) + \sigma_n^2.$$

其中

$$[\bar{K}_{nm}]_{ij} = K(\boldsymbol{x}_i, \bar{\boldsymbol{x}}_j),\ \Lambda = \mathrm{diag}(\boldsymbol{\lambda}),$$

$$\boldsymbol{\lambda}_n = K_{nn} - \bar{K}_{nm}^{\rm T}\bar{K}_{mm}^{-1}\bar{K}_{mn},$$

$$\bar{Q}_{mm} = \bar{K}_{mm} + \bar{K}_{mn}(\Lambda + \sigma_n^2 I)^{-1}\bar{K}_{nm}.$$

在 SPGP 法的模型训练中, 除了要学习超参数外, 还要确定伪数据集的最优位置, 共有 $md + num(\theta)$ 个参数. 这些未知参数一般通过梯度上升法来最大化边缘似然函数.

由于 $m \ll n$, SPGP 法的计算效率得到大幅提高, 训练计算量为 $O(m^2 n)$, 单测试样本的均值和协方差预测的复杂度分别为 $O(m)$ 和 $O(m^2)$.

表 1 归纳了上述几种 GPR 近似方法的计算量.

上述近似方法也称全局 GPR 近似法, 这是因为这些方法试图利用所选的包含集来表征所有 n 个数据点. 另一种不同的近似方法即是局部 GPR 近似法 —— 仅利用测试点附近的训练数据点集用于预测; 当一个变化比较剧烈的数据集 (如研究对象函数曲线

表 1　标准 GPR 和近似 GPR 的计算量比较 ($m < n$)

计算量	训　练	单个测试样本	
		均值预测	协方差预测
标准 GPR	$O(n^3)$	$O(n)$	$O(n^2)$
SD 近似法	$O(m^2n)$	$O(m)$	$O(m^2)$
Nyström 法	$O(m^2n)$	$O(n)$	$O(mn)$
SR 近似法	$O(m^2n)$	$O(m)$	$O(m^2)$
PP 近似法	$O(m^2n)$	$O(m)$	$O(m^2)$
SPGP 近似法	$O(m^2n)$	$O(m)$	$O(m^2)$

严重振荡等) 难以用一个小数据子集(包含集) 表征时, 局部 GPR 近似法能给出一个更快、更精确的结果. Snelson 等[16]结合全局 GPR 法和局部 GPR 法的优点, 提出了一种新的稀疏 GPR 近似法 —— 部分独立条件 (PIC) 近似法.

各种各样的近似方法仍在不断涌现, 如稀疏在线高斯过程[17]、增量在线稀疏法[18]以及进化高斯过程[19]等. 为了提高 GRP 法的效率, 可以采用硬件如图形处理器 (GPU) 等并行处理技术[20].

2.2　突破高斯噪声分布假设的改进方法

由第 1 节的 GPR 方法原理可知, 存在一个假设 —— 噪声必须满足高斯分布, 即观测数据满足多变量联合高斯分布. 该假设使得 GPR 方法中的矩阵运算变得简单方便, 其预测分布也满足高斯型. 但是许多实际情况并不满足这个假设, 如观测值为正且在好几个数量级之间变化时, 这种情形难以直接假设一个同方差的高斯噪声. 一般做法是先对其进行取对数 log 变换处理, 然后假设变换后的数据受高斯噪声污染, 此时 GPR 方法能得到较好的效果. 实际中, 存在一些其他连续变换, 可以把观测空间的数据转换到某一个能够用 GPR 方法很好建模的空间, log 变换即是其中的一种. 基于这种思想, Snelson 等[21]提出了翘曲高斯过程 (WGP) 方法.

假定 $\boldsymbol{z}$ 为真实观测矢量经过同一单调函数 t 映射转换到隐式空间的隐式观测值矢量, 即 $\boldsymbol{z}$ 中每一个元素满足

$$z = t(y;\, \Psi). \tag{5}$$

应用 GPR 方法对 $\boldsymbol{z}$ 进行回归, 可得 $p(\boldsymbol{z}|\boldsymbol{\theta}) = N(0, C)$, 其中 $\boldsymbol{\theta}$ 和 C 定义如前. 易得负对数似然函数 $L_{\boldsymbol{z}}$ 为

$$L_{\boldsymbol{z}} = -\log p(\boldsymbol{z}|\boldsymbol{\theta}) = \frac{1}{2}\log\det C + \frac{1}{2}\boldsymbol{z}^{\mathrm{T}}C^{-1}\boldsymbol{z} + \frac{n}{2}\log(2\pi).$$

再应用式 (5), 可得

$$L = -\log p(\boldsymbol{y}|\boldsymbol{\theta},\, \Psi) = \frac{1}{2}\log\det C + \frac{1}{2}\boldsymbol{t}(\boldsymbol{y})^{\mathrm{T}}C^{-1}\boldsymbol{t}(\boldsymbol{y}) - \sum_{i=1}^{n}\log\left.\frac{\partial t(y)}{\partial y}\right|_{y_i} + \frac{n}{2}\log(2\pi), \tag{6}$$

其中 $\boldsymbol{t}(\boldsymbol{y}) = [t(y_1), t(y_2), \cdots, t(y_n)]^{\mathrm{T}}$.

与 GPR 训练原理一样, WGPR 模型的训练也是通过式 (6) 对参数 Ψ 和 $\boldsymbol{\theta}$ 求偏导, 再采用共轭梯度法等优化方法对偏导数进行最小化得到参数的最优解. 可以看出, 超参数 $\boldsymbol{\theta}$ 和非线性翘曲函数的优化是同时进行的. 同 GPR 预测原理, 可得新测试样本 $\boldsymbol{x}_*$ 对应预测值 z_* 的后验分布为

$$z_*|\boldsymbol{y}, \boldsymbol{\theta}, \Psi \sim N(\mu_*^z, (\sigma_*^z)^2).$$

其中

$$\mu_*^z = K(\boldsymbol{x}_*, X)[K(X,X) + \sigma_n^2 I_n]^{-1}\boldsymbol{z} = K(\boldsymbol{x}_*, X)[K(X,X) + \sigma_n^2 I_n]^{-1}\boldsymbol{t}(\boldsymbol{y}),$$

$$(\sigma_*^z)^2 = k(\boldsymbol{x}_*, \boldsymbol{x}_*) - K(\boldsymbol{x}_*, X)\times [K(X,X) + \sigma_n^2 I_n]^{-1}K(X, \boldsymbol{x}_*) + \sigma_n^2.$$

在真实观测空间中, 预测后验分布变为

$$p(f_*|\boldsymbol{y}, \boldsymbol{\theta}, \Psi) = \frac{t'(f_*)}{\sqrt{2\pi(\sigma_*^z)^2}}\exp\left[-\frac{1}{2}\left(\frac{t(f_*) - \mu_*^z}{\sigma_*^z}\right)^2\right].$$

由上式可以看出, 预测后验分布的形状取决于翘曲函数 t, 一般为非对称且多峰值的. 一种可选的翘曲函数为如下双曲正切函数的神经网络式求和:

$$t(y;\, \Psi) = y + \sum_{i=1}^{I} a_i \tanh(b_i(y + c_i)),\ a_i, b_i \geqslant 0,\ \forall i,$$

其中 $\Psi = \{\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}\}$.

此外, 目前通常采用共轭梯度法求取训练样本对数似然函数的极大值以自适应地获得最优超参数, 但是共轭梯度法存在优化效果初值依赖性强、迭代次数难以确定、易陷入局部最优解的缺点. 针对这种情况, 刘开云等[22]采用十进制遗传算法代替共轭梯度法搜寻高斯过程最优超参数, 有效避免了共轭梯度法的缺陷, 可以在参数搜索区间快速找到全局最优解, 从而提高 GPR 的泛化性能. Zhu 等[23]利用粒子群算法优化超参数并用于位移预报, 得到的预报精度优于遗传算法. 申倩倩等[24]提出了在 GP 的训练中使用自适应自然梯度法, 即基于自适应自然梯度法的在线 GPR 建模算法, 满足了在线建模算法的实时性要求. Petelin 等[25]经实验研究验证了 3 种随机优化方法 (遗传算法、差分进化算法和粒子群算法) 用于超参数优化的有效性.

3　与神经网络、支持向量机的关系

随着机器学习领域研究的不断深入, 许多基于机器学习的先进算法已广泛地应用于非线性回归、分类、概率密度估计和数据挖掘等领域, 例如神经网络和支持向量机, 其在解决回归和分类问题中已取得了一定成果. 然而, 神经网络在研究过程中通常都存在如何选择一个合适的网络架构, 如何从数据中获取更

多的有用信息等问题; 支持向量机则存在如何选取合适的惩罚项来防止过拟合, 如何确定核函数参数以及如何定量评价预测输出等问题.

从贝叶斯的观点来看, 神经网络方法可以视为在非线性函数簇上定义一个先验概率分布, 其学习过程可用未知函数上的后验概率分布来描述 (如一些学习算法是以最大化后验概率来获取最优函数, 一些蒙特卡罗方法是从该后验概率分布中采样的等). Buntine等[26]、MacKay[27-28]和Neal[29]等几乎同时提出将贝叶斯分析方法和神经网络相结合, 在网络权重空间中充分考虑其概率分布, 先验分布经过贝叶斯推理得到后验分布, 这点与一般的神经网络设计方法明显不同. Neal[30]于1996年发现, 当神经网络的隐层节点数趋于无穷大时, 网络权重的高斯先验分布将趋近于一个GP, 神经网络模型的超参数决定了GP的参数. 该发现促使研究人员从研究参数化神经网络方法转向更为直接的GP方法; 此时,神经网络中的参数优化计算变为GP中协方差矩阵的简单矩阵运算. Williams等[31]于1996年提出将GP方法推广到原本由神经网络、决策树等方法所解决的高维回归问题中.

统计学习是机器学习的一种实现方法, Vapnik等[32]从20世纪六七十年代就开始了这方面的研究. 随着统计学习理论的不断发展, 产生了许多基于统计学习理论体系的通用机器学习方法, 其中支持向量机和高斯过程都是基于统计学习理论发展起来的核学习机, 对于处理高维数、小样本以及非线性等复杂问题具有很好的适应性, 且泛化能力强. 而高斯过程应用了贝叶斯技巧, 得到的模型属于非参数概率模型, 其优势主要表现在:

1) 不仅能够对未知输入进行预测输出, 而且能够对该预测输出的精度参数或不确定性 (即估计方差) 进行定量分析;

2) 能以先验概率的形式表示过程的先验知识, 并通过标准的贝叶斯方法进行模型选择, 从而提高了过程模型性能;

3) 与神经网络、支持向量机等方法相比, 其模型参数明显减少, 且能方便地推断出超参数.

4 高斯过程回归方法的发展及应用

虽然高斯过程在20世纪90年代中期才开始应用于机器学习领域[33], 但是基于高斯过程的预测, 特别是对于时间序列分析而言, 其基础理论至少可追溯到20世纪40年代[34-35], 例如统计地质学中的"Kriging法"[36-37]即为高斯过程预测, 该方法先后在空间预测[38]和空间统计[39-40]上得到了应用. 至此, 人们逐渐意识到高斯过程回归可以用于解决一般的回归问题. 文献[41-43]利用一系列计算机仿真实验验证了GP方法的有效性, 并讨论了超参数优化等问题. Williams等[1,31]基于机器学习理论系统地阐述了GP方法的基本原理及应用, 将GP方法的应用推向了一个新的高度.

4.1 用于时间序列预测分析

GPR方法在时间序列预测分析中的应用历史较为悠久, 近年来又不断地得到了发展和完善. Brahim-Belhouari等[44-45]应用GPR方法对平稳和非平稳时间序列进行了预测研究. Girard等[46]基于GPR方法解决了输入不确定情况下时间序列的多步预测问题. Zhang等[47]提出了一种用于时间序列分析的高效率GPR方法. Wang等[48]对比分析了人工神经网络 (ANN) 和GPR方法在时间序列预测上的应用效果, 指出GPR方法更适合于非平稳情形. Farrell等[49]应用GPR方法进行了股票趋势预测. 在国内, 苏国韶和徐冲等将GPR方法分别应用于基坑非线性位移时间序列预测[50]、隧道围岩变形预报[51]、隧道位移时序分析和边坡变形预测[52].

4.2 用于动态系统模型辨识

GPR方法因其独特的优势, 自20世纪90年代末便开始应用于动态系统模型的辨识. Murray-Smith等[53]基于蒙特卡罗方法对高斯过程先验模型和多模型方法进行了分析比较. Gregorčič[54]针对参数化多模型方法存在结构确定难、参数获取困难以及"维数灾难"等不足, 将高斯过程用于动态非线性系统的建模, 对输入空间维数选择和多步预测等问题进行探讨, 给出了模型结构的选择方法, 并应用于液压系统. Ni等[55]针对大多数工业过程中存在的非线性和时变特性严重削弱了传统软传感器预测性能的问题, 提出了基于双重更新和双重预处理两个策略的移动窗GPR方法, 并应用于动态非线性系统辨识, 有效提高了对动态过程的跟踪性能.

Lawrence[56]对GP进行了拓展, 提出一种新的非线性隐变量模型 —— 高斯过程隐变量模型 (GP-LVM). Wang等[57]在隐空间内应用GP-LVM对动态系统模型进行辨识. 此外, 王磊等[6]应用高斯过程对表情动作单元进行跟踪, 并利用高斯过程隐变量空间的分布方差对跟踪效果实施有效约束, 降低了跟踪过程中的非数值型误差.

4.3 用于系统控制或控制系统设计

GPR方法能够给出预测值的不确定度, 因此能方便地与预测控制、自适应控制等方法相结合, 由此出现了一系列预测控制和自适应控制等新方法.

Kocijan[58]于 2002 年率先将 GPR 模型提供的方差信息引入控制信号的优化过程, 提出了一种新的预测控制方法. Likar 等[59]建立了气液分离装置的 GPR 模型, 并基于此模型实现了预测函数控制. 此外, 基于 GPR 模型的预测控制方法还有很多, 如内模控制方法[60]、随机预测控制方法[61]等, 在实际应用中都取得了很好的效果.

Murray-Smith 等[62]于 2002 年将 GPR 模型引入自适应控制过程, 所得控制器能够自适应地跟踪参考信号和从观测响应中学习系统模型. 针对非最小相位非线性系统, Sbarbaro 等[63]结合 GPR 模型, 设计了一种自适应非参数控制器. Rottmann 等[64]基于 GPR 模型, 分开并交替学习系统的动态模型和价值函数, 提出了一种自适应自冶控制方法, 能够实时学习系统的控制策略, 并成功地应用于微型飞船高度的实时控制. Petelin 等[19]提出了进化 GP 模型, 并基于此模型实现了自适应控制系统的设计.

其他基于 GPR 模型的控制算法参见文献 [2].

4.4　与贝叶斯滤波方法相结合

传统的滤波方法大多要求系统模型和先验噪声统计特性已知, 然而在实际中难以精确获取系统模型和噪声统计特性, 导致滤波方法的性能受限甚至无法正常工作. GPR 模型能够提供预测值的不确定度, 使其能够方便地与滤波方法相结合, 可以在一定程度上克服滤波方法对系统模型和噪声统计特性的依赖性.

Ferris 等[65]于 2006 年最先将 GPR 与高斯滤波相结合, 提出了高斯过程粒子滤波 (GP-PF), 并在基于无线电强度估计的移动载体定位中得到了应用. 随后, Ko、Deisenroth 等先后提出了高斯过程扩展卡尔曼滤波 (GP-EKF)[66]、高斯过程 Unscented 卡尔曼滤波 (GP-UKF)[67]以及高斯过程假设密度滤波算法[68]. 2008 年, Ko 等[69]提出了高斯过程滤波这一概念, 并对相关算法进行了总结, 通过实验验证了高斯过程滤波性能的优越性. 近年来, 李鹏等[5,70-71]将高斯过程回归融入平方根 UKF 算法中, 提出一种不确定系统模型协方差自适应调节滤波算法, 并将其应用于无人飞行器 SINS/GPS 组合导航和航天器交会对接过程中.

5　展望与结论

与神经网络和支持向量机相比, GPR 方法具有容易实现、灵活的非参数推断、超参数自适应获取等优点, 是一个具有概率意义的核学习机, 可对预测输出做出概率解释, 在实际应用中已取得了许多令人满意的成果. 但是, 目前 GPR 方法还不够完善, 仍在不断地发展, 主要有以下几个发展趋势[2,72]:

1) 计算量大是限制 GPR 方法应用的主要问题, 寻求效率更高的协方差求逆计算方法或训练集选择方法仍是不变的研究内容. 一方面, 可以结合计算机软硬件及并行计算技术, 提高计算效率; 另一方面, 自动处理数据并寻找“信息数据”以压缩数据集来降低计算量是另一发展趋势. 此外, 基于 GPR 模型的递归辨识或在线学习方法的高效实现方法仍面临着一些挑战.

2) 对于控制系统而言, 抗干扰性能至关重要, 但是目前大部分基于 GPR 模型的控制方法更多地仅关注设定点的跟踪性能, 缺少关于抗干扰的性能分析和设计. 另外, 基于 GPR 模型的鲁棒控制设计也将是今后研究的趋势之一.

3) 利用 GPR 方法辨识动态系统的状态方程和观测方程, 有效解决了滤波过程中由于模型不准确或统计特性未知而导致滤波结果发散的问题, 优势明显, 可以与更多滤波方法 (如容积卡尔曼滤波等) 相结合, 并应用于实际工程中.

随着贝叶斯理论和统计学习理论的进一步深入发展以及计算技术的飞速进步, 日趋成熟完善和不断实用化的 GPR 方法将不断拓宽其应用领域, 如生物系统等不确定系统模型辨识等; 而新应用、新要求也将促使 GPR 方法不断取得新的进展.

参考文献(References)

[1] Williams C K I, Rasmussen C E. Gaussian processes for machine learning[M]. Cambridge: MIT Press, 2006: 7-32.

[2] Kocijan J. Control algorithms based on Gaussian process models: A state-of-the-art survey[C]. Proc of the Special Int Conf on Complex Systems: Synergy of Control, Communications and Computing. Ohrid, 2011: 69-80.

[3] Park C W, Huang J H Z, Ding Y. Domain decomposition approach for fast Gaussian process regression of large spatial data sets[J]. J of Machine Learning Research, 2011, 12: 1697-1728.

[4] He Z K, Liu G B, Zhao X J, et al. Temperature model for FOG zero-bias using Gaussian process regression[J]. Advances in Intelligent Systems and Computing, 2012, 180: 37-45.

[5] 李鹏, 宋申民, 段广仁. 改进的平方根 UKF 及其在交会对接中的应用[J]. 电机与控制学报, 2010, 14(11): 100-104.
(Li P, Song S M, Duan G R. Improved square root unscented Kalman filter and its application in rendezvous and docking[J]. Electric Machines and Control, 2010, 14(11): 100-104.)

[6] 王磊, 邹北骥, 彭小宁, 等. 基于高斯过程的表情动作单元跟踪技术[J]. 电子学报, 2007, 35(11): 2087-2091.
(Wang L, Zou B J, Peng X N, et al. Facial tracking

by Gaussian process[J]. Acta Electronica Sinica, 2007, 35(11): 2087-2091.)

[7] Snelson E. Flexible and efficient Gaussian process models for machine learning[D]. London: Gatsby Computational Neuroscience Unit, University of London, 2007.

[8] Williams C K I, Seeger M. Using the Nyström method to speed up kernel machines[C]. Proc of the Int Conf on Advances in Neural Information Processing Systems(NIPS) 13. Denver, 2001: 682-688.

[9] Wahba G. Spline models for observational data[M]. Philadelphia: Society for Industrial and Applied Mathematics, 1990: 95-100.

[10] Poggio T, Girosi F. Networks for approximation and learning[J]. Proc of IEEE, 1990, 78(9): 1481-1497.

[11] Williams C K I, Rasmussen C E, Schwaighofer A, et al. Observations on the Nyström method for Gaussian process prediction[C]. Proc of the NIPS 12. Denver, 2000: 464-473.

[12] Seeger M, Williams C K I, Lawrence N D. Fast forward selection to speed up sparse Gaussian process regression[C]. Proc of the 9th Int Workshop on Artificial Intelligence and Statistics. Florida, 2003: 1-8.

[13] Keerthi S, Chu W. A matching pursuit approach to sparse Gaussian process regression[C]. Proc of the NIPS 18. Vancouver, 2005: 643-650.

[14] Tresp V. A Bayesian committee machine[J]. Neural Computation, 2000, 12: 2719-2741.

[15] Snelson E, Ghahramani Z. Sparse Gaussian processes using pseudo-inputs[C]. Proc of the NIPS 18. Vancouver, 2006: 1257-1264.

[16] Snelson E, Ghahramani Z. Local and global sparse Gaussian process approximations[C]. Proc of the 11th Int Workshop on Artificial Intelligence and Statistics. Puertorico, 2007: 524-531.

[17] Csató L, Opper M. Sparse online Gaussian processes[J]. Neural Computation, 2002, 14(3): 641-668.

[18] Nguyen-Tuong D, Peters J. Incremental online sparsification for model learning in realtime robot control[J]. Neurocomputing, 2011, 74(11): 1859-1867.

[19] Petelin D, Kocijan J. Control system with evolving Gaussian process model[C]. Proc of IEEE Symposium Series on Computational Intelligence. Paris, 2011: 178-184.

[20] Musizza B, Petelin D, Kocijan J. Accelerated learning of Gaussian process models[C]. Proc of the 7th EUROSIM Congress on Modelling and Simulation. Praha, 2010: 8-14.

[21] Snelson E, Rasmussen C E, Ghahramani Z. Warped Gaussian processes[C]. Proc of the NIPS 16. Vancouver, 2004: 337-344.

[22] 刘开云, 刘保国, 徐冲. 基于遗传-组合核函数高斯过程回归算法的边坡非线性变形时序分析智能模型[J]. 岩石力学与工程学报, 2009, 28(10): 2128-2134.
(Liu K Y, Liu B G, Xu C. Intelligent analysis model of slope nonlinear displacement time series based on genetic-Gaussian process regression algorithm of combined kernel function[J]. Chinese J of Rock Mechanics and Engineering, 2009, 28(10): 2128-2134.)

[23] Zhu F W, Xu C, Dui G S. Particle swarm hybridize with Gaussian process regression for displacement prediction[C]. Proc of the 5th IEEE Int Conf on Bio-Inspired Computing: Theories and Applications. Changsha, 2010: 522-525.

[24] 申倩倩, 孙宗海. 基于自适应自然梯度法的在线高斯过程建模[J]. 计算机应用研究, 2011, 28(1): 95-97.
(Shen Q Q, Sun Z H. Online learning algorithm of Gaussian process based on adaptive nature gradient[J]. Application Research of Computers, 2011, 28(1): 95-120.)

[25] Petelin D, Filipič B, Kocijan J. Optimization of Gaussian process models with evolutionary algorithms[C]. Proc of the 10th Int Conf on Adaptive and Natural Computing Algorithms. Slovenia, 2011: 420-429.

[26] Buntine W, Weigend A. Bayesian back propagation[J]. Complex Systems, 1991, 5(6): 603-643.

[27] MacKay D. A practical Bayesian framework for backprop networks[J]. Neural Computation, 1992, 4(3): 448-472.

[28] MacKay D. Bayesian methods for neural networks: Theory and applications[R]. Cambridge: Cavendish Lab, Cambridge University, 1995.

[29] Neal R M. Bayesian training of backpropagation networks by the hybrid Monte Carlo method[R]. Toronto: Graduate Department of Computer Science, University of Toronto, 1993.

[30] Neal R M. Bayesian learning for neural networks[D]. Toronto: Graduate Department of Computer Science, University of Toronto, 1995: 34-60.

[31] Williams C K I, Rasmussen C E. Gaussian processes for regression[C]. Proc of the NIPS 8. Denver, 1996: 514-520.

[32] Vapnik V N. Statistical learning theory[M]. New York: John Wiley & Sons, 1998: 267-288.

[33] Rasmussen C E. Evaluation of Gaussian processes and other methods for non-linear regression[D]. Toronto, Graduate Department of Computer Science, University of Toronto, 1996.

[34] Wiener N. Extrapolation, interpolation and smoothing of stationary time series[M]. Cambridge: MIT Press, 1949: 33-61.

[35] Kolmogorov A N. Interpolation and extrapolation of stationary random sequences[J]. Bulletin of the Academy

of Science of the USSR, Series in Mathematic, 1941, 5(1): 3-14.

[36] Matheron G. The intrinsic random functions and their applications[J]. Advances in Applied Probability, 1973, 5(3): 439-468.

[37] Journel A G, Huijbregts C J. Mining geostatistics[M]. New York: Academic Press, 1978: 387-396.

[38] Whittle P. Prediction and regulation by linear least-square methods[M]. London: English Universities Press, 1963: 5-31.

[39] Ripley B. Spatial statistics[M]. New York: Wiley, 1981: 179-224.

[40] Cressie N A C. Statistics for spatial data[M]. New York: Wiley, 1993: 115-356.

[41] O'Hagan A. Curve fitting and optimal design for prediction[J]. J of the Royal Statistical Society B, 1978, 40(1): 1-42.

[42] Sacks J, Welch W J, Mitchell T J, et al. Design and analysis of computer experiments[J]. Statistical Science, 1989, 4(4): 409-435.

[43] Santner T J, Williams B J, Notz W. The design and analysis of computer experiments[M]. New York: Springer, 2003: 81-158.

[44] Brahim-Belhouari S, Vesin J M. Bayesian learning using Gaussian process for time series prediction[C]. Proc of the 11th IEEE Workshop on Statistical Signal Processing. Singapore, 2001: 433-436.

[45] Brahim-Belhouari S, Bermak A. Gaussian process for nonstationary time series prediction[J]. Computational Statistics & Data Analysis, 2004, 47(4): 705-712.

[46] Girard A, Rasmussen C E, Quiñonero Candela J, et al. Gaussian process priors with uncertain inputs — Application to multiple-step ahead time series forecasting[C]. Proc of the NIPS 15. Vancouver, 2003: 529-536.

[47] Zhang Y N, Leithead W E, Leith D J. Time-series Gaussian process regression based on Toeplitz computation of $O(N_2)$ operations and $O(N)$-level storage[C]. Proc of the 44th IEEE Conf on Decision and Control, and the European Control Conf 2005. Seville, 2005: 3711-3716.

[48] Wang T D, Chuang S J, Fyfe C. Comparing Gaussian processes and artificial neural networks for forecasting[C]. Proc of 9th Joint Conf on Information Sciences. Taiwan, 2006: 1-4.

[49] Todd Farrell M, Correa A. Gaussian process regression models for predicting stock trends[J]. Relation, 2007, 10(1): 3414.

[50] 苏国韶, 燕柳斌, 张小飞, 等. 基坑位移时间序列预测的高斯过程方法[J]. 广西大学学报, 2007, 32(2): 223-226.
(Su G S, Yan L B, Zhang X F, et al. Time series prediction of foundation pit displacement using Gaussian process method[J]. J of Guangxi University, 2007, 32(2): 223-226.)

[51] 苏国韶, 张研, 燕柳斌. 隧道围岩变形预报的高斯过程机器学习模型[J]. 桂林理工大学学报, 2010, 30(4): 551-555.
(Su G S, Zhang Y, Yan L B. Deformation forecasting for tunnel rock by Gaussian process machine learning model[J]. J of Guilin University of Technology, 2010, 30(4): 551-555.)

[52] 徐冲. 分岔隧道设计施工优化与稳定性评价[D]. 北京: 北京交通大学土木建筑工程学院, 2011: 23-56.
(Xu C. Study on optimization and stability evaluation of design and construction of forked tunnel[D]. Beijing: School of Civil Engineering, Beijing Jiaotong University, 2011: 23-56.)

[53] Murray-Smith R, Johansen T A, Shorten R. On transient dynamics, off-equilibrium behaviour and identification in blended multiple model structures[C]. Proc of the European Control Conf. Karslruhe, 1999: BA-14.

[54] Gregorčič G, Lightbody G. Gaussian processes for modelling of dynamic non-linear systems[C]. Proc of the Irish Signals and Systems Conf. Cork, 2002: 141-147.

[55] Ni W D, Tan S K, Ng W J, et al. Moving-window GPR for nonlinear dynamic system modeling with dual updating and dual preprocessing[J]. Industrial and Engineering Chemistry Research, 2012, 51(18): 6416-6428.

[56] Lawrence N D. Gaussian process latent variable models for visualisation of high dimensional data[C]. Proc of the NIPS 16. Vancouver, 2004: 329-336.

[57] Wang J, Fleet D, Hertzmann A. Gaussian process dynamical models[C]. Proc of the NIPS 18. Vancouver, 2006: 1441-1448.

[58] Kocijan J. Gaussian process model based predictive control[R]. Ljubljana: Institute Jožef Stefan, 2002.

[59] Likar B, Kocijan J. Predictive control of a gas-liquid separation plant based on a Gaussian process model[J]. Computers and Chemical Engineering, 2007, 31(3): 142-152.

[60] Gregorčič G, Lightbody G. Internal model control based on Gaussian process prior model[C]. Proc of the 2003 American Control Conf. Denver, 2003: 4981-4986.

[61] Grancharova A, Kocijan J. Stochastic predictive control of a thermoelectric power plant[C]. Proc of the Int Conf on Automatics and Informatics. Sofia, 2007: I-13-I-16.

[62] Murray-Smith R, Sbarbaro D. Nonlinear adaptive control using nonparametric Gaussian process prior models[C]. Proc of the 15th IFAC World Congress. Barcelona, 2002: 1038-1043.

(下转第1137页)